Rapport d’analyse

Dans le cadre du projet de master 1 d’informatique de l’année 2015-2016 de l’université de La Rochelle, il nous est demandé, en considérant comme client M. Revel Arnaud, de réaliser un système permettant le suivi et la gestion des admissions de l’université, le projet ayant pour but de concrétiser les acquis des cours. Le projet a pour principales problématiques l’accessibilité par internet, et le respect des responsabilités dans les actions sur les dossiers et des limites de temps imposé par la loi.

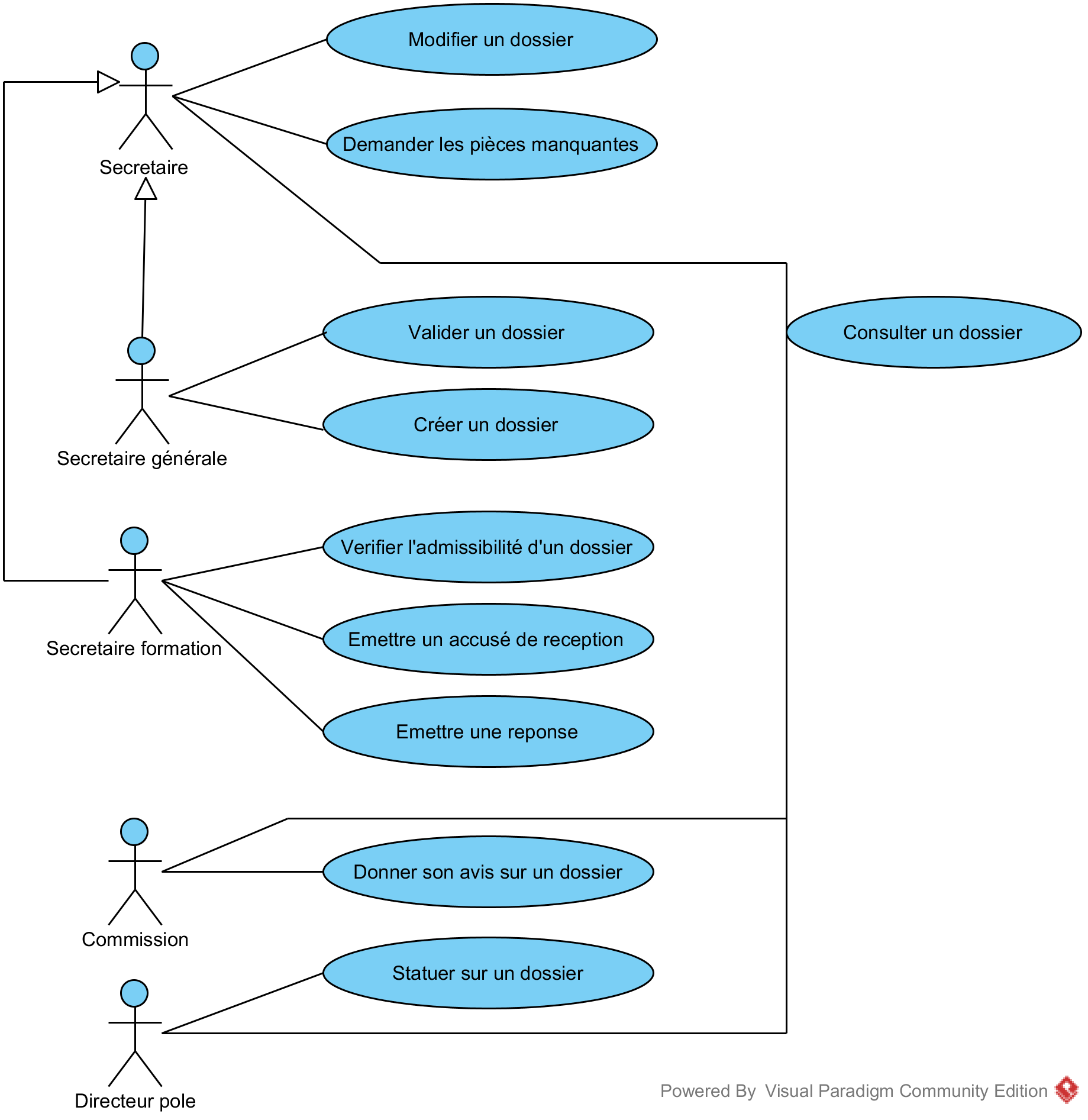
Ainsi le document suivant présente les résultats de l’analyse de ce projet.

**1. Exigences et contraintes**

Synthèse des exigences et des contraintes donnée dans le cahier des charges.

|  |  |
| --- | --- |
| Exigences | Contraintes |
| - Gérer la gestion des dossiers de candidature à l’ULR  - Gérer la création du dossier  - Permettre l'accès via internet  - Gérer les différentes connexions  - Gérer les droits en fonction des différents  niveaux de responsabilités  - Permettre d’avoir un lien entre les différents services  - Permettre la mise en place de commissions de recrutements avec un délai d’environ mois entre chaque commission  - S’assurer qu’une réponse est envoyée dans les 2 mois qui suivent l’accusé de réception  - Permettre un suivi des différentes étapes du dossier  - Permettre de gérer les différentes formations | - S’assurer que les commissions rentrent dans les dates (dernière semaine du mois)  - Permettre l’inscription à partir de mars (Date fixée par l’administrateur)  - Empêcher de s’inscrire après la mi-juin (Date  fixée par l’administrateur)  - Vérifier qu’il y ait au moins l’INE, l’adresse, le nom et le prénom de l’étudiant avant la validation de la première étape  - Permettre la génération automatique d’une lettre de refus ou d’acceptation une fois que le dossier est traité pour permettre l’envoi d’une réponse  - Trier par niveau de langue pour les candidatures étrangères (minimum niveau B2)  - Empêcher de passer à une étape sans avoir validé les étapes antérieures |

**2. Cas d’utilisation**



**Acteurs :**

Secrétaire : classe abstraite répertoriant les actions accessible par les deux types de secrétaire, soit modifier un dossier, et demander les pièces manquantes d’un dossier.

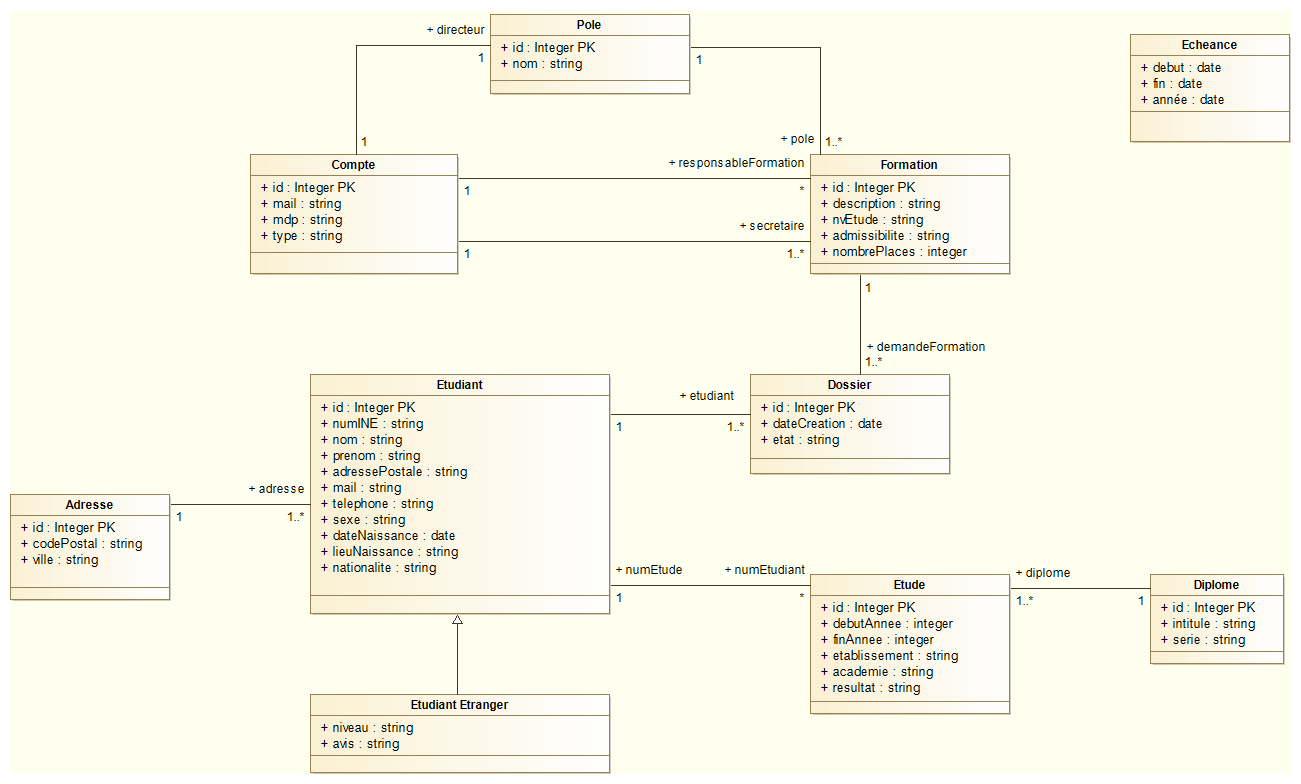
Secrétaire générale : se charge de créer les dossiers puis de les valider.

Secrétaire formation : secrétaire responsable d’une formation. Emet un accusé de réception, vérifie l’admissibilité si besoin, puis après retour de la décision du directeur émet la réponse pour le postulant.

Commission : donne son avis sur l’acceptation d’une admission.

Directeur de pole : directeur responsable d’un pôle. Valider ou refuse les admissions.

**3. Modèle du domaine**



Compte : contient les informations des comptes permettant de se connecter au système. L’attribut « type » permet de différencier le type de compte, et ainsi de gérer les différentes possibilités selon les responsabilités du connecté.

Pole : répertorie les différent pole et contient le compte du directeur du pole.

Formation : contient les différentes formations proposées avec le pole concerné, le compte du responsable de formation et de la secrétaire responsable de la formation, et les informations décrivant la formation.

Dossier : contient la date de création, et l’état actuel du dossier pour un étudiant donné avec une formation demandé.

Etudiant : contient les données correspondant à un étudiant.

Etudiant Etranger : héritant d’Etudiant, on y ajoute les informations nécessaires pour un étudiant étranger (le niveau en français, et pour ceux venant de campus France, l’avis de campus France).

Adresse : contient les villes et le code postal correspondant. Permet d’éviter la redondance d’information stockée.

Etude : contient les informations demandées pour les dernières études suivies.

Diplôme : contient les différents diplômes qu’un étudiant peut avoir obtenu. Permet d’éviter la redondance d’information stockée.

Echeance : permet de stockée la plage de date permettant l’admission pour une année donnée.

**4. IHM**