DOCUMENT DE CONCEPTION

Introduction

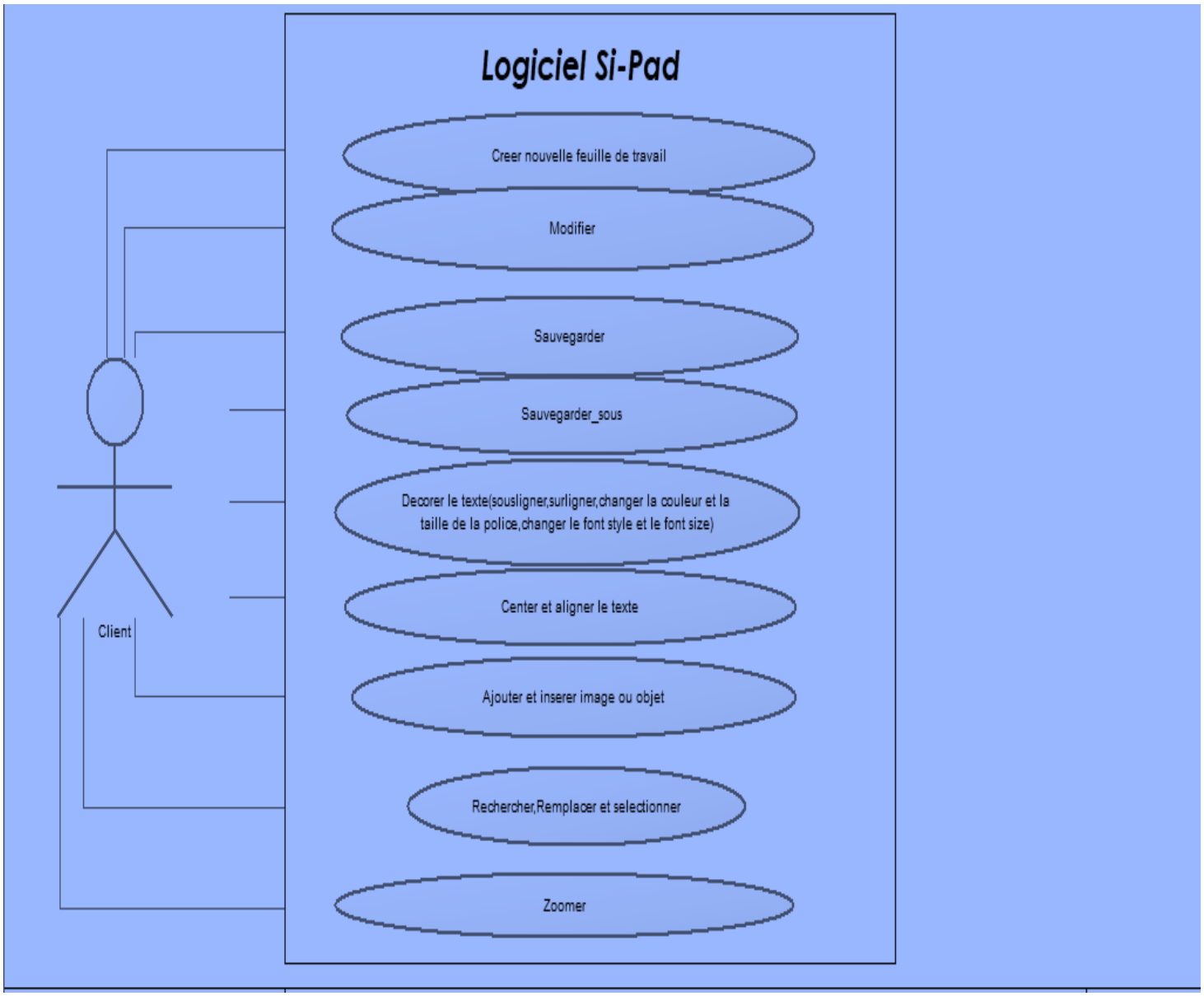
Ce présent document contient l'architecture globale de notre logiciel. Il est diviser en deux grandes parties .Dans la première parties nous donnerons le diagramme des cas d'utilisation et le diagramme de classe relatif à notre logiciel et dans la deuxième partie nous parlerons de l'interface, des normes de codage et de conception de ce logiciel puis nous donnerons les contraintes de conception.

I.Diagrammes

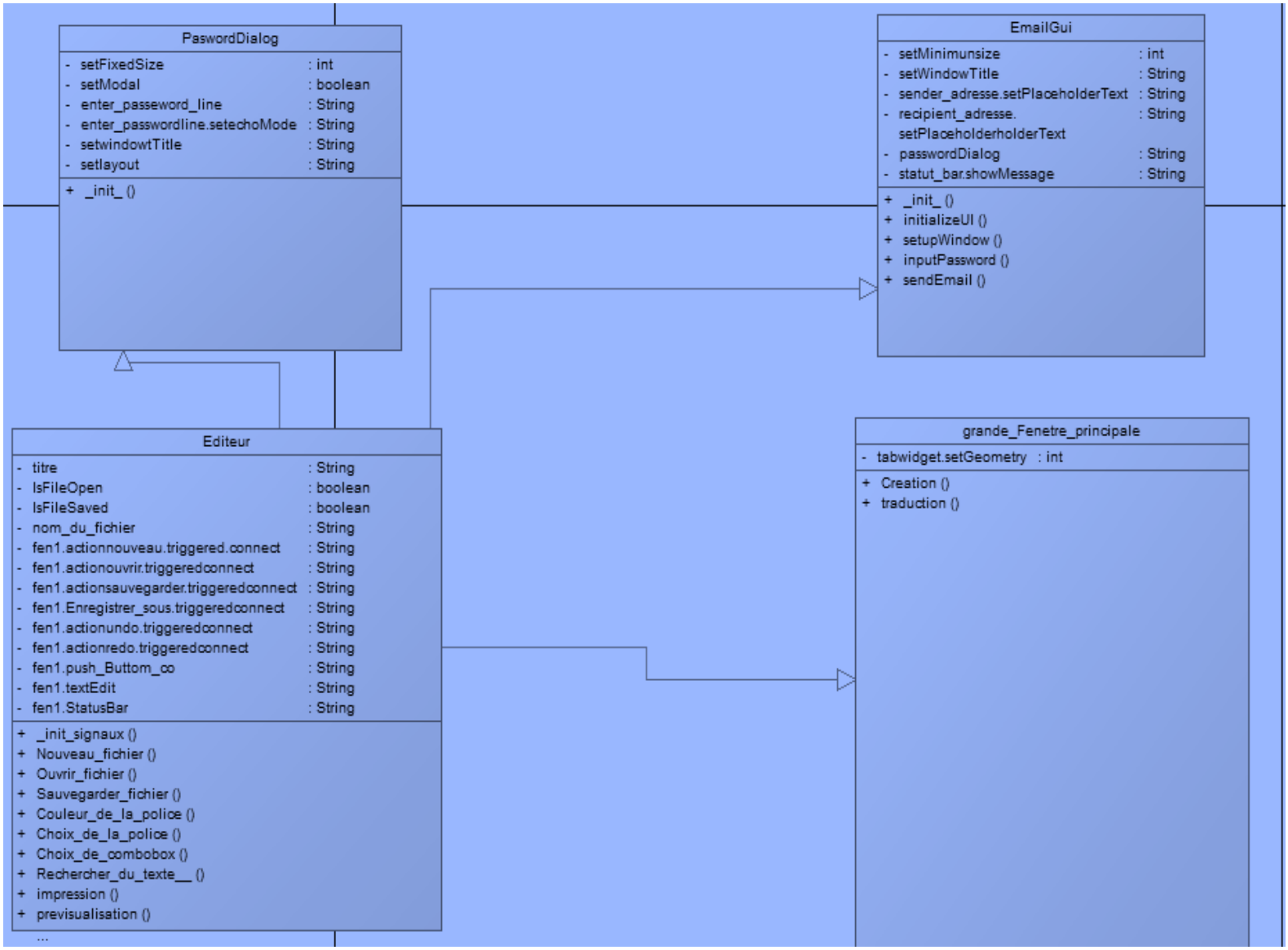
I.1-Diagramme des cas d'utilisation

Nous savons que le logiciel que nous sommes amenés à produire est un éditeur de texte du style WordPad .Cependant à quoi servira-t-il concrètement ?La réponse à

cette question est donnée par le diagramme des cas d’utilisation donnée ci-dessous.



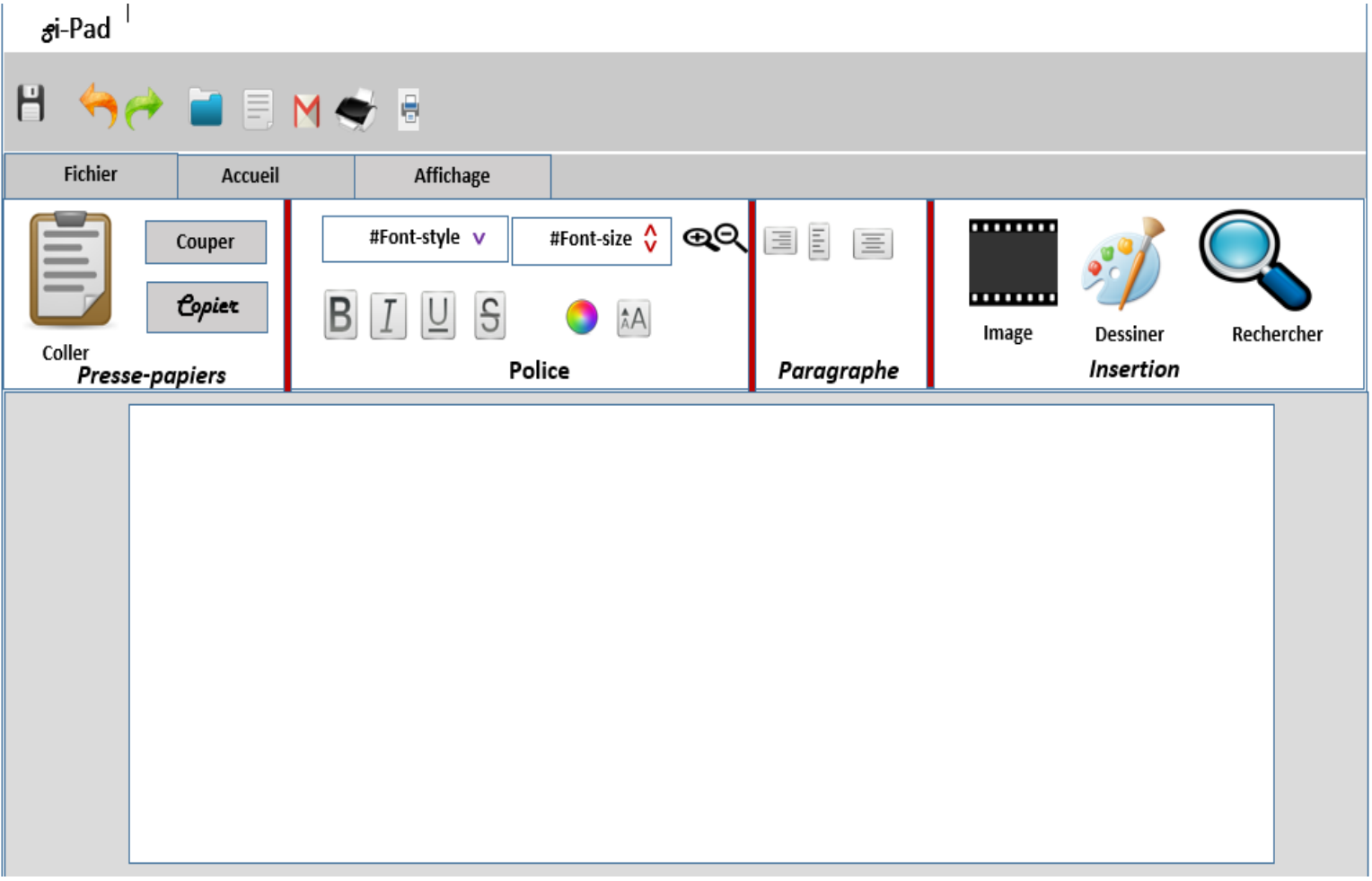
I.2-Diagrammes de classes



II-Interfaces, Normes et Contraintes de conception

II.1-Interface utilisateur

L'interface utilisateur de notre logiciel Si-Pad aura cet aspect de manière générale (Il pourrait éventuellement avoir quelque modification. Cependant on aura a peu prêt le logiciel aura a peu prêt le même aspect).



II.2-Interface logiciel

L'interface de notre logiciel ou encore son logo est le suivant :

II.3-Norme de conception du logiciel

La représentation graphique de la conception du logiciel doit être fait à partir de diagramme UML (diagramme de classe, diagramme des cas d'utilisation).Dans le cadre de ce projet ces diagrammes ont étés fait à partir du logiciel PowerAMC.

II.4-Normes de codage logiciel

Le logiciel Si-Pad doit être fait à partir du langage de programmation python. Il devra donc respecter les règles de codage relatives à ce langage.

Appelation

Le nom des classes et des méthodes ne doit pas contenir de caractères accentués, il ne doit pas contenir de caractère spécial, s'il contient plus de deux mot ces mots devront être séparé par des Under scores(_) .En résumé le nom des classes et des méthodes doit être du type alphanumérique et pourra éventuellement contenir des caractères en majuscule. Cette règle est valable également pour les attributs.

Documentation

Pour la bonne compréhension du code source il devra être conçu de la sorte :

-Chaque class doit porter le nom de l'élément qu'elle est sensé représenter et son rôle doit être mis en commentaire.

-chaque méthode doit porter le nom de la fonctionnalité qui lui est associe Ex : ***def Nouveau_fichier pour produire la méthode permettant créer un nouveau fichier.***

-Le nom des attributs doivent être choisis en fonction de ce qu'ils sont sensé représenter .Ils devrons être en français ou en anglais selon le bon vouloir du développeur.

II.5-Contraintes de conception

Pour la réalisation de ce projet nous devrons utiliser le langage de programmation python ainsi que les modules compatibles avec ce langage. Il devra utiliser le concept de programmation orientée objet i.e. utiliser la notion de classes.