

# Cahier des Charges

## A) Présentation du Client

Ce projet ayant lieu dans un cours de développement informatique, il s'agit ici d'une situation un peu particulière où les étudiants sont à la fois les créateurs, mais également les clients. De plus, les professeurs supervisant l'ensemble du projet sont également client de ce dernier car ce sont eux qui ont fixé le travail à remettre.

## B) Présentation du Projet

Les élèves trouvent que le dossier de téléchargements de leurs ordinateurs n'est pas facilement lisible et utilisable, en effet, bien que celui-ci permette un tri des fichiers en fonction de leurs extensions, on ne peut néanmoins pas dire que ce dossier soit des plus ergonomiques. Etant un dossier régulièrement utilisé au quotidien, les élèves ont ainsi eu l'idée de créer un programme permettant dans un premier de juste trier et créer des dossiers regroupant un type de fichier dans ce dossier de téléchargement.

Au-delà de l'aspect ergonomique et fonctionnel de cette nouvelle mise en page du dossier, les utilisateurs pourront ainsi plus facilement accéder à tel ou tel fichier, car ils n'auront qu'à accéder au dossier portant le nom de l'extension recherchée et ainsi gagner un temps qui peut paraître négligeable sur le moment mais qui au fil du temps pourra s'avérer assez conséquent.

Les élèves implémenteront également par après d'autres outils permettant à l'utilisateur de gagner un temps précieux, et de visualiser plus facilement les actions à effectuer vis-à-vis de ces différents fichiers.

## C) Objectif du Client

L'objectif principal des clients est bien évidemment de gagner un maximum de temps en facilitant la lecture et en optimisant l'affichage de son dossier de téléchargement. Afin de consacrer plus de temps à leur travail.

## D) Intervenants

Les clients sont ici à la fois les prestataires mais également les professeurs premier clients de ce programme.

Les prestataires sont ici Arthur Schamroth et Firmin Moyen, deux élèves en deuxième année de technologie de l'informatique à la Haute Ecole Ephec de Louvain-la-Neuve.

## E) Cible/Utilisateurs

Le programme est destiné à être utilisé par toute personne voulant en disposer, mais il sera néanmoins dans un premier temps exclusif aux clients, ici donc, les professeurs et les prestataires du projet.

## F) Demandes fonctionnelles

- 1) Une fois le programme lancé, le dossier de téléchargements sera divisé en différents sous dossiers portant chacun un nom d'extension de fichiers présents dans le dossier de téléchargements.

- 2) Une fois le programme lancé, les fichiers seront déplacés automatiquement dans leur sous-dossier respectif.
- 3) Dans un menu présent en console, une comparaison de la taille d'un des sous-dossiers à l'ensemble des fichiers présents dans le dossier de téléchargement sera proposée.
- 4) Dans ce menu en console ainsi que dans un interface graphique, l'utilisateur aura un choix lui permettant de choisir une extension de fichier afin d'obtenir une liste de fichiers de ce type ainsi qu'une brève description de celui-ci.
- 5) Dans un menu d'interface graphique, l'utilisateur aura le choix de lancer n'importe quel fichier dans Google Chrome.
- 6) Dans une menu d'interface graphique, l'utilisateur aura la possibilité de regarder l'évolution de leur fichier de téléchargements dans le temps.

## G) Contraintes

Les clients désirent que le programme soit codé en python. Pour ce qui est du reste, aucune contrainte n'est réellement imposée aux prestataires, qui sont en fin de compte assez libre dans leur programme, ainsi s'ils désirent utiliser une base de données, aucune ne leur est imposée par exemple.

Néanmoins, un minimum de documentation dans le code du programme est préférable afin que celui-ci soit compréhensible par quiconque voudrait y jeter un œil.

## H) Charte graphique/ergonomie

Les clients désirent un style simple et ergonomique quant à l'interface graphique dans lequel ceux-ci devront écrire leur extension désirée. Ils ne désirent pas particulièrement devoir avoir à faire à la console, en effet, l'utilisateur devra juste écrire une extension spécifique dans un champ pour que celui-ci retourne l'ensemble des fichiers répondant à cette extension, le reste se déroulera uniquement à la souris, donc pas besoin de console.

## I) Enveloppe budgétaire

Ce projet ayant lieu dans le cadre d'un cours de développement de l'informatique, il est évident que les prestataires ne seront pas rémunérés pour leur travail.

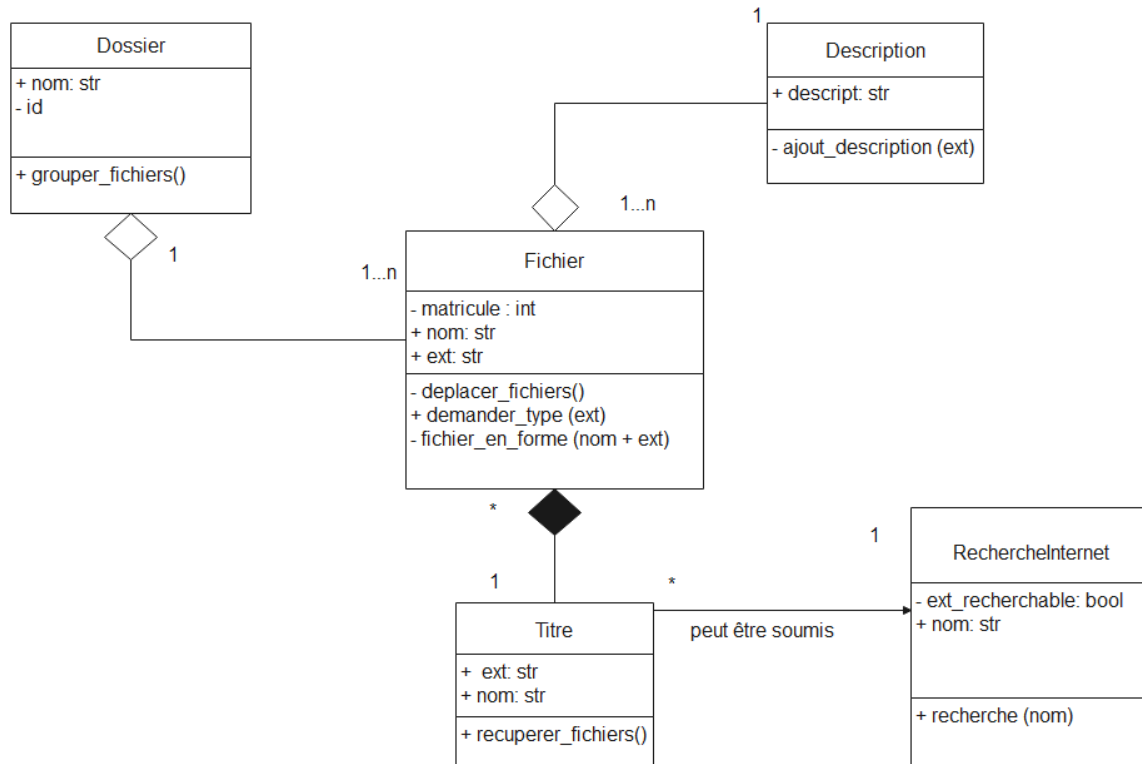
## J) Planification

Le projet est prévu sur une période de plus ou moins deux mois, l'élaboration de celui-ci commencera dès que le MVP déjà soumis sera validé par les professeurs/clients.

Les prestataires devront remettre pour la semaine du 09/11 le MVP complet de leur projet, afin de démontrer le bon commencement et avancement de leur projet.

Pas d'autre planification n'est demandée par les clients pour l'instant.

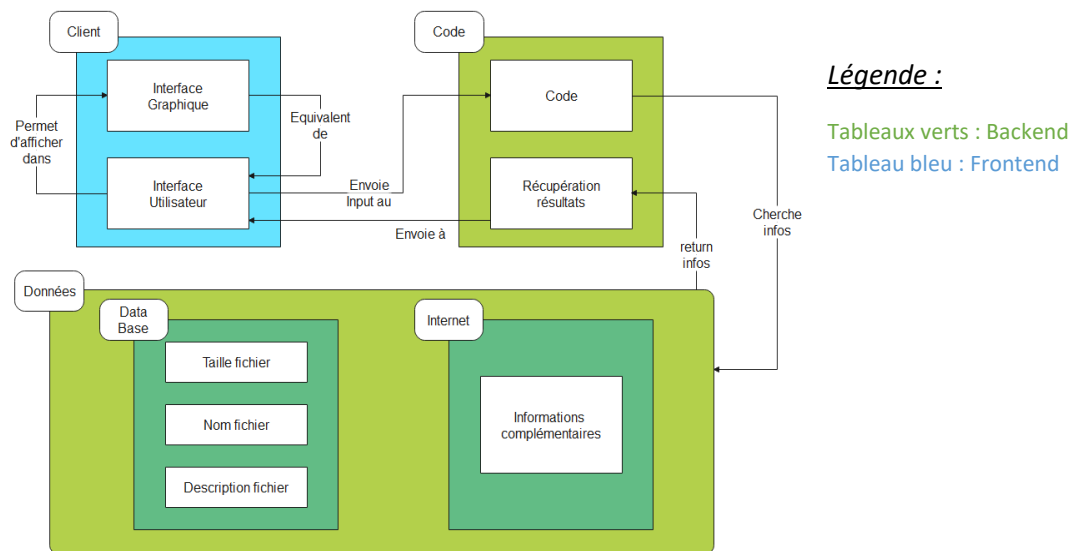
## K) Diagramme de classe



Dans le diagramme présenté ci-dessus, on retrouve un tableau « Titre » qui correspond au nom du « Fichier », ce titre est composé d'un nom et d'une extension, tous deux présentés sous la forme d'un string. Ces titres sont également susceptibles d'être le sujet d'une recherche internet dans le cadre de notre programme, mais tout dépendra de leur attribut « ext », en effet si celui-ci correspond à un « mp4 » il n'y a aucun intérêt à effectuer une recherche en ligne d'un fichier mp4.

Les prestataires décident ensuite d'attribuer à chaque « Fichier » un matricule unique afin de mieux les identifier et pouvoir ainsi les utiliser plus facilement dans le futur. Ce « Fichier » est à la base des « Dossiers », en effet, c'est grâce à son attribut « nom » que nous pourrions nommer ces derniers, ces dossiers posséderont également un id unique permettant de les identifier et les utiliser également plus facilement, comme par exemple dans une base de données. Ces fichiers posséderont également une description propre en fonction de leur « nom », ce nom qui rappelons-le est directement lié à l'extension du tableau « Titre ».

## L) Diagramme d'architecture



Dans le diagramme ci-dessus, on retrouve un espace « client » composé d'une interface graphique et d'un interface utilisateur, l'interface graphique a un aspect plus « user-friendly » que l'interface utilisateur qui est souvent une console ou invite de commande. C'est dans cet interface graphique que l'utilisateur complètera un champ input où il mettra l'extension du type de fichier qu'il désire obtenir, dans ce projet, l'interface graphique possède la même fonction que l'interface utilisateur.

Une fois que le client a introduit un type de fichier précis, le champ input est directement au code, afin d'aller récupérer certaines informations dans notre stockage de données. Tous les fichiers ayant déjà été récupéré et traité lors de l'exécution du code s'y trouvent déjà, nous retrouvons dans notre base de données une multitude d'informations intéressantes pour la suite du programme. Néanmoins, la base de données n'est pas la seule source de données intéressantes pour ce programme, en effet l'utilisateur a la possibilité d'effectuer une recherche internet concernant le type de fichier sélectionné, et ce, depuis l'interface graphique directement.

Prenons un exemple, si dans l'affichage graphique, l'utilisateur demande à obtenir tous les fichiers txt du dossier de téléchargements, le programme lui retournera dans un premier temps la liste de ceux-ci mais offrira également la possibilité à l'utilisateur de supprimer un fichier, après l'avoir sélectionné ou d'obtenir une description de ce type de fichier, pour cela nous passons donc par notre base de données. En revanche, si l'utilisateur décide d'effectuer une recherche internet directement depuis l'interface graphique, le programme utilisera donc internet et non plus notre base de données.