Rapport de veille technologique et de choix stratégique du groupe

Cadrage du besoin :

Pour le parseur d'articles scientifiques en format texte, nous allons avoir besoin de choisir un langage de programmation. Celui choisi par notre groupe sera Python. Python a été choisi car il est un langage de programmation haut-niveau avec un très grand nombre de bibliothèques spécialisées.

Nous allons devoir faire attention aux problèmes de conversion d'images ou de schéma ainsi qu'aux caractères spéciaux qui peuvent être potentiellement difficiles à convertir.

Dans le déroulement de ce projet il y aura 5 acteurs :

- Kessler Rémy (client et représentant du laboratoire de l'IRISA)
- Casagrande Donovan (maître SCRUM)
- Maurice Benoît (membre de l'équipe SCRUM)
- Gros Arthur (membre de l'équipe SCRUM)
- Haurogne Benjamin (membre de l'équipe SCRUM)

Les axes de recherches sont l'utilisation de bibliothèques efficaces et le choix d'un algorithme ou d'un code efficient.

Sourcing :

La librairie Python utilisée pour l'analyse de pdf est PyPDF2. Une autre librairie sera essayée pour comparer la meilleure des 2 bibliothèques (pdfminer).

Nous disposons de 2 outils libres dont nous nous en inspirerons : « pdftotext » et « pdf2txt » en majorité.

Collecte de l'information :

Pour répondre aux différents besoins, la majorité des informations nous viennent de l'utilisation de « pdftotext » et « pdf2txt ».

Analyse de l’information :

« pdftotext » a comme qualité de prendre en compte le contenu des schémas et de les afficher au bon endroit mais si le schéma est complexe il rend le document incompréhensible, et il affiche certains caractères spéciaux (ex : µ, ξ, ≤, …).

Il a comme défaut de ne pas espacer correctement les paragraphes, les titres et les tableaux, n’affiche pas sur la même ligne le numéro et le titre de la partie.

« pdf2txt » a comme qualité d’espacer correctement les différentes parties (ex : mettre un “\n” entre 2 paragraphes), affiche correctement un tableau en espaçant les différentes cases du tableau, et affiche certains caractères spéciaux (ex : µ, ξ, ≤, …).

Il a comme défaut d’afficher le contenu des schémas qu’à la fin de la page, et n’affiche pas certains caractères spéciaux (ex : µ, ξ, ≤, …).

Les deux outils ont comme points communs de ne pas prendre en compte la police, la taille, la couleur du texte et les formes géométriques. Ils ne prennent en compte que les caractères.

Diffusion de l’information :

Afin de garantir l’accès à l’information à tous les acteurs du projet, un GitHub a été créé.

Des rendus seront envoyés via le Moodle au client.

Des réunions hebdomadaires sont organisées entre le client (Kessler Rémy) et le maître SCRUM (Casagrande Donovan) ainsi qu’entre le maître SCRUM et son équipe (Maurice Benoît, Gros Arthur et Haurogne Benjamin).