

Sprint 2 Conception logicielle - Scrum 2025



Parseur d'articles scientifiques en format texte

Dans la séance précédente vous avez choisi le convertisseur le plus adapté PDF → texte. Il s'agit maintenant d'analyser les articles plain texte par le parseur. Il sera développé dans un langage adéquat, et analysera les sorties textuelles du convertisseur.

Nous voulons comme sortie un fichier **.txt** par fichier ayant les sections et la structure suivante:

Le nom du fichier d'origine (dans une ligne, espaces convertis à _)

Le titre du papier (dans une ligne)

Le résumé (abstract) de l'auteur (dans une ligne)

Considérez que les sections ne sont pas forcément toutes présentes dans les papiers.

Le système doit prendre comme entrée un dossier contenant les fichiers PDF et créer un sous-dossier pour déposer les sorties en plain texte avec le même noms que les PDF. Si le sous-dossier existe, il faudra l'effacer.

Groupe (Torres)

Dépôt : avant sprint 3

Le langage doit être choisi en fonctions des critères :

- 1/ Maîtrisé par les membres de l'équipe
- 2/ Performance en temps d'exécution

Suggestions pour la performance : mesurer avec **time** sur un programme avec boucles imbriquées de 0 à 10000 ; faire l'ouverture et lecture de gros fichiers remplis de random, etc. Testez plusieurs langages, puis votez et décidez.

Le système final et les versions intermédiaires doivent être placés sur Github ; chaque version dans une branche différente. Le fichier README.md doit contenir une explication générale du système et la procédure nécessaire pour lancer les programmes.

Il faut documenter les événements/artefacts Scrum en ajoutant sur l'ENT des photos de votre travail (réunions de planification de sprint, mêlée quotidienne, revue de sprint, rétrospective du sprint) et déposer sur ENT par le maître SCRUM.

Consignes

- Équipes et rôles : fixés en séance 1
- Méthodologie agile : SCRUM
- Plateforme : le projet doit fonctionner sous GNU/Linux en ligne de commande
- Aucune conversion en ligne via navigateur n'est acceptée
- Langage : Perl (5 ou 6), Python (2, 3), C/C++, Ruby ou un mix de ces langages.
- Soyez agiles !



Vous êtes en compétition ! Attention aux fuites des idées ou de logiciel !