1 Description

Le but de ce projet est de fabriquer un logiciel qui parcourt un site web en suivant tous les liens et télécharge tous les fichiers HTML. Une fois l'exploration (le parcours de toutes les URL) effectuée, le logiciel doit permettre de chercher un mot clé.

Le projet doit présenter à l'utilisateur une interface web minimaliste qui permet de :

— Lancer l'exploration sur un site donné.

— Une fois l'exploration terminée affiche la liste des fichiers HTML, images et autre fichier trouvé.

— Une fois l'exploration terminée permet d'effectuer la recherche d'un mot clé et affiche les pages contenant ce mot clé.

Pour effectuer une recherche par mot clé, un index des mots doit être construit et géré. L'index des mots est une structure de données qui à chaque mot rencontré sur une page web associe la liste des URL ou il a été trouvé. Cette structure est remplie en parcourant les parties de texte des fichiers HTML, il faudra ignorer les balises HTML, le code JavaScript et le code CSS.

2 Structure

Le projet comporte deux parties, un *serveur web* qui fournit l'interface utilisateur et des *processus explorateurs* qui vont effectuer l'exploration d'une URL. Le serveur va également coordonner les processus.

2.1 Explorateurs

Les explorateurs se connectent à un serveur à leur création puis attendent des requêtes de celui-ci. Lorsqu'une requête est reçue sous forme d'une URL à explorer, l'explorateur télécharge le fichier HTML associé puis cherche dans le code HTML toutes les URL contenues dans le fichier.

Si l'URL correspond à un fichier qui n'est pas de type HTML (identifié grâce à l'en tête "Content-Type"), le fichier n'est pas téléchargé.

Une fois que le fichier a été exploré, le résultat est envoyé au serveur. Le résultat contient deux parties :

— La liste de toutes les URL contenues dans le fichier dans le cas d'un fichier HTML. Du type et de la taille du fichier dans les autres cas.

— La liste de tous les mots rencontrés dans le fichier HTML à ajouter à l'index.

2.2 Serveur

Le serveur a plusieurs rôles :

1. Il fournit un serveur web minimaliste qui affiche une page web à l'utilisateur demandant de saisir une URL. Une fois la requête calculée, il renvoie le résultat à l'utilisateur sous forme de page web.
2. Pour chaque requête d'un utilisateur, il envoie la requête à un des explorateurs connecté.

3. Il coordonne les différents explorateurs en implémentant une sorte d'algorithme de parcours de graphe :
   Il contient un ensemble des URL et leurs états : nouvelle ; en cours d'exploration ; déjà exploré. Quand un explorateurs lui envoie un résultat, il effectue les actions suivantes
   — Marque l'URL qu'il a explorée comme déjà explorée ;
   — Ajoute les URL du résultat comme de nouvelles URL sauf si elles sont déjà connues ;
   — Choisi une nouvelle URL la marque comme en cours d'exploration et effectue une requête à un processus explorateur pour qu'il l'explore.
   Pour garantir que le programme s'arrête, vous pouvez faire de sorte que, si une URL correspond un fichier en dehors du site en cours d'exploration, elle n'est pas explorée. Vous pouvez aussi mettre un limite aux nombre d'URL à explorer avant de s'arrêter.
4. Il maintient l'index des mots : À chaque fois qu'un résultat est reçu, il ajoute les mots et l'URL associée à l'index.
5. Le résultat des requêtes est sauvegardé dans un fichier qui est lu au démarrage pour éviter de faire des explorations inutiles.

3 Consignes Techniques

En utilisant Java NIO, vous devez implémenter vous-même :
— Toutes les requêtes au site web et le téléchargement des fichiers HTML.
— La gestion des sérialisations entre vos clients et votre serveur,
Pour semplifier la tâche, vous pouvez utiliser la classe HTTPURLConnection uniquement dans le serveur pour l'interface utilisateur, pas dans les processus explorateurs.

Le projet sera réalisé en binôme.

Le projet devra être rendu sur git, des comptes vous seront créés sur `git-etudiants.lacl.fr`. Des commit réguliers (au minimum 2 par semaines) sont attendus.

Je vous encourage à faire beaucoup des tests sur des sites web différents, les comportements des serveurs varient. Malheureusement vous ne pourrez pas vous connecter au site en HTTPS.

4 Suggestion de feuille de route

Je vous suggère de travailler sur le projet dans l'ordre suivant :

1. Réaliser un explorateur simplifié sans la construction de l'index. Ce processus fait l'exploration d'une URL et écrit la liste des URL sur sa sortie standard. Une URL peut être trouvée dans le code HTML en cherchant des chaînes de caractère de la forme : `href="..."` la partie entre guillemets étant l'URL.
2. Réfléchisser au protocole entre votre client et votre serveur et implémenter le code de sérialisation nécessaire
3. Réaliser un serveur qui fait l'exploration d'un site web fixe (L'URL est passée par la ligne de commande ou dans écris dans le code) sans la partie d'index et de sauvegarde et qui écrit le résultat sur la sortie standard.
4. Réaliser la partie serveur web sans la partie d'index et de sauvegarde.
5. Implementer la recherche.
6. Implementer la sauvegarde.

5 Dates

— Choix des binômes sur Eprel avant le 6 mars.
— Dernier commit mercredi 1 avril minuit. De courtes soutenances auront lieu le 2 ou 3 avril.