

Plan de mémoire

Refonte du moteur de recherche du Parlement Européen

- 1 Statut et mission de l'auditeur au sein du Parlement Européen
- 2 La situation initiale
 - 2.1 Présentation et fonction de la GSA
 - 2.2 Les lacunes du système
 - 2.2.1 Une fonction principale insatisfaisante
 - 2.2.2 Des fonctions secondaires absentes
 - 2.3 Une échéance à relever
- 3 Les contraintes techniques du Parlement
 - 3.1 Les technologies supportées
 - 3.2 Un projet pilote vers le Cloud
- 4 Cahier des charges fonctionnelles
 - 4.1 Les besoins de la direction de la communication
 - 4.2 Les besoins de la direction technique
- 5 Cahier des charges techniques
 - 5.1 Les technologies autorisées
 - 5.2 Les processus qualité
 - 5.2.1 Le suivi (framework ASTRA)
 - 5.2.2 Documentation
 - 5.2.3 Intégration continue (Jenkins)
 - 5.2.4 Tests unitaires et couverture de code
 - 5.2.5 Revue de code automatique avec Sonar
- 6 Les étapes de la réalisation du projet
 - 6.1 Une première phase d'analyse
 - 6.1.1 Définition de l'architecture applicative
 - 6.1.2 Etat de l'art des solutions techniques applicables
 - 6.2 Une phase de recherche et de validation de la solution technique
 - 6.2.1 Un POC du module d'indexation
 - 6.2.2 La solution Azur Search
 - 6.2.3 Les solutions Amazon
 - 6.2.3.1 CloudSearch
 - 6.2.3.2 Elasticsearch service
(Parler de la gestion des droits)
 - 6.2.4 L'hébergement des index Elasticsearch sur le réseau du Parlement
 - 6.2.5 Conclusion
 - 6.3 Mise en place de la solution retenue
 - 6.3.1 Mise à jour de l'architecture applicative
(ajout d'un web service de recherche)
 - 6.3.2 Un développement par itération
 - 6.3.2.1 Une technologie mal comprise
(note : par la maîtrise d'ouvrage et par la maîtrise d'œuvre)
 - 6.3.2.2 De plus prioritaire au moins prioritaire
 - 6.3.2.3 Des phases d'analyse approfondie et une redéfinition des besoins

- 6.3.2.4 Validation et boucle de retro-action
(note : exemples :
fréquence parsing trop importante,
synonymes trop laborieux remplacé par la suggestion de phrase,
autocompletion fulltext remplacée par l'historique des requêtes)
- 6.3.3 Finition du module d'indexation et de son interface d'administration
 - 6.3.3.1 La gestion des fréquences d'indexation
 - 6.3.3.2 Le lien entre les pages html et le modèle de données
 - 6.3.3.3 La distinction entre les liens suivis et les pages indexées
 - 6.3.3.4 Politiques d'indexation et suppression des pages obsolètes
 - 6.3.3.5 La gestion des pages en erreur
 - 6.3.3.6 Une interface d'édition des index
- 6.3.4 L'implémentation des fonctionnalités de recherches
 - 6.3.4.1 Résumé des concepts Elasticsearch
 - 6.3.4.2 Le couplage entre la structure des index et les fonctionnalités de recherche
 - 6.3.4.3 La recherche sur une phrase
 - 6.3.4.4 Une recherche simultanée sur la phrase et sur les mots
 - 6.3.4.5 Corrections et suggestions de « phrases »
 - 6.3.4.6 Auto complétion sur l'historique des requêtes pertinentes
- 6.3.5 Le web service de recherche
 - 6.3.5.1 Un web service REST utilisable depuis n'importe quelle source
 - 6.3.5.2 Une interface qui masque la complexité
 - 6.3.5.3 La définition du modèle de donnée exposé
 - 6.3.5.4 Implémentation d'une page de recherche simplifiée
- 6.3.6 Intégration du nouveau web service de recherche au site internet du parlement
- 6.4 La documentation
 - 6.4.1 Le Document Technique d'Architecture (DTA)
 - 6.4.2 La documentation technique
 - 6.4.3 La documentation Java et les commentaires
- 6.5 La formation utilisateur
- 7 Retours d'expérience
- 8 Conclusion